1. What is ML

➔ Machine learning (ML) is defined as a discipline of artificial intelligence (AI) that provides machines the ability to automatically learn from data and past experiences to identify patterns and make predictions with minimal human intervention.

2. What are different ML algorithms

➔ Linear regression.

- Logistic regression.
- Decision tree.
- SVM algorithm.
- Naive Bayes algorithm.
- KNN algorithm.
- K-means.
- Random forest algorithm.

3. What is linear regression

➔ In statistics, linear regression is a linear approach for modelling the relationship between a scalar response and one or more explanatory variables. The case of one explanatory variable is called simple linear regression; for more than one, the process is called multiple linear regression.

4. What is Logistic Regression

➔ **Logistic regression** is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). Like all regression analyses, logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.

5. Explain working of KNN

➔ K-nearest neighbors (KNN) algorithm uses 'feature similarity' to predict the values of new datapoints which further means that the new data point will be assigned a value based on how closely it matches the points in the training set. We can understand its working with the help of following steps −

Step 1 − For implementing any algorithm, we need dataset. So during the first step of KNN, we must load the training as well as test data.

Step 2 − Next, we need to choose the value of K i.e. the nearest data points. K can be any integer.

Step 3 − For each point in the test data do the following −

- **3.1** − Calculate the distance between test data and each row of training data with the help of any of the method namely: Euclidean, Manhattan or Hamming distance. The most commonly used method to calculate distance is Euclidean.
- **3.2** − Now, based on the distance value, sort them in ascending order.
- **3.3** − Next, it will choose the top K rows from the sorted array.
- **3.4** − Now, it will assign a class to the test point based on most frequent class of these rows.

Step 4 − End

6. Explain working of K-mean Clustering

➔ **Step-1:** Select the number K to decide the number of clusters.

Step-2: Select random K points or centroids. (It can be other from the input dataset).

Step-3: Assign each data point to their closest centroid, which will form the predefined K clusters.

Step-4: Calculate the variance and place a new centroid of each cluster.

Step-5: Repeat the third steps, which means reassign each datapoint to the new closest centroid of each cluster.

Step-6: If any reassignment occurs, then go to step-4 else go to FINISH.

Step-7: The model is ready.

7. What is get_dummies

➔ The **get_dummies** function is used to convert categorical variables into dummy or indicator variables.

A dummy or indicator variable can have a value of 0 or 1.

The get_dummies function works as follows:

- It takes a data frame, series, or list.
- Then, it converts each unique element present in the object to a column heading.
- The function iterates over the object that is passed and checks if the element at the particular index matches the column heading.
- If it does, it encodes it as a 1.
- Otherwise, it assigns it a 0.

8. How K is calculated in K-mean

➔ 1. **Elbow Curve Method**

The elbow method runs k-means clustering on the dataset for a range of values of k (say 1 to 10).

- Perform K-means clustering with all these different values of K. For each of the K values, we calculate average distances to the centroid across all data points.
- Plot these points and find the point where the average distance from the centroid falls suddenly ("Elbow").

2. Silhouette analysis

The silhouette coefficient is a measure of how similar a data point is within-cluster (cohesion) compared to other clusters (separation).

- Select a range of values of k (say 1 to 10).
- Plot Silhouette coefficient for each value of K.

The equation for calculating the silhouette coefficient for a particular data point:

$$S(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

- S(i) is the silhouette coefficient of the data point i.
- a(i) is the average distance between i and all the other data points in the cluster to which i belongs.
- b(i) is the average distance from i to all clusters to which i does not belong.

9. What is recursion, give example

➔ Recursion means "defining a problem in terms of itself". This can be a very powerful tool in writing algorithms. Recursion comes directly from Mathematics, where there are many examples of expressions written in terms of themselves. For example, the Fibonacci sequence is defined as: **F(i) = F(i-1) + F(i-2)**

10. What is Greedy method, Dynamic programming

➔ Greedy programming is the approach that tries to solve a problem as quickly as possible, while dynamic programming is the approach that tries to solve a problem as efficiently as possible.

In greedy programming, you try to solve a problem as quickly as possible by trying to find the smallest possible solution. In dynamic programming, you try to solve a problem as efficiently as possible by trying to find the smallest possible solution.

11. Compare greedy and Dynamic programming

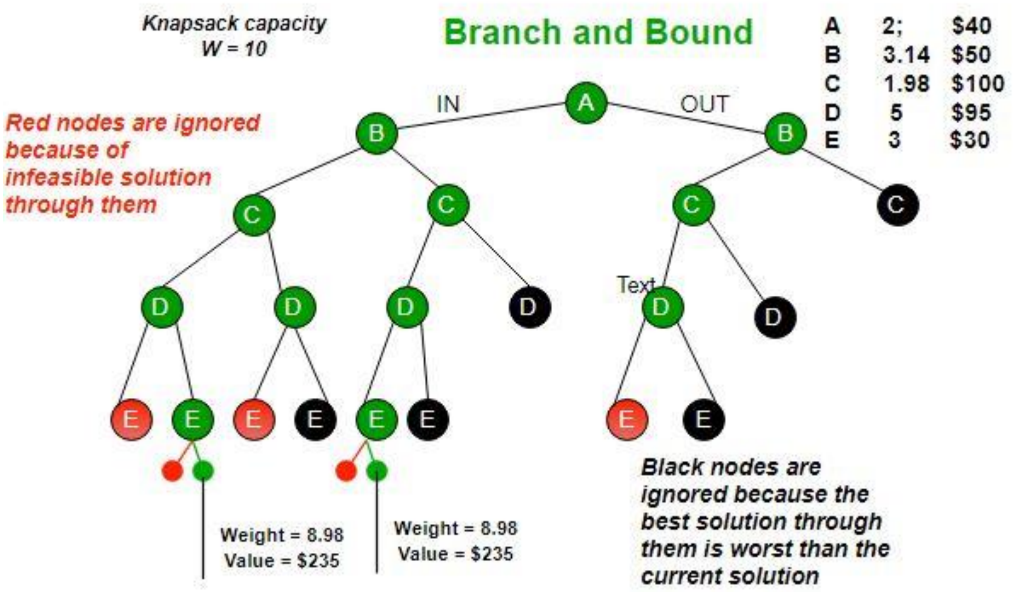
➔

Greedy Programming	**Dynamic Programming**
A greedy algorithm chooses the best solution at the moment, in order to ensure a global optimal solution.	In dynamic programming, we look at the current problem and the current solution to determine whether to make a particular choice or not. We then calculate the optimal choice based on previous problems and solutions.
It is not guaranteed that an optimal solution will be obtained in the greedy method.	Because of the nature of Dynamic Programming, it is certain that an optimal solution will be generated.
A greedy methodology follows the problem-solving approach of making the locally optimal choice at each step.	Dynamic programming is an algorithmic approach that uses a recurring formula to calculate new states.
The greedy approach deterministically obtains its answer by repeatedly selecting a random step in a backward direction and never looking back or changing previous choices.	Developing a solution top down or bottom up is accomplished by obtaining smaller optimal sub-solutions.
Fractional knapsack is an example of greedy algorithms.	0/1 knapsack problem is an example of greedy algorithms.
Every problem can't be solved by greedy algorithm.	Every problem can be solved by Dynamic algorithm.
A solution to a specified problem set is contained within the given solution set.	It is not necessary to insist on a particular set of feasible solutions.
More Efficient because we never look back to other options.	Less Efficient as compared to a greedy approach becausee it's required DP table to store the answers of calculated states.
A greedy strategy is faster than a dynamic one.	Compared to greedy programming, it is slower.
Fast results	Slow results comparatively
Each step is locally optimal.	Past solutions are used to create new ones.

12. What is branch and bound, give example

➔ Branch and bound is one of the techniques used for problem solving. It is similar to the backtracking since it also uses the state space tree. It is used for solving the optimization problems and minimization problems. If we have given a maximization problem then we can convert it using the Branch and bound technique by simply converting the problem into a maximization problem.

Let's see the Branch and Bound Approach to solve the **0/1 Knapsack problem**: The Backtracking Solution can be optimized if we know a bound on best possible solution subtree rooted with every node. If the best in subtree is worse than current best, we can simply ignore this node and its subtrees. So we compute bound (best solution) for every node and compare the bound with current best solution before exploring the node.
Example bounds used in below diagram are, A down can give $315, B down can $275, C down can $225, D down can $125 and E down can $30.


Knapsack capacity
W = 10
Branch and Bound
IN
OUT
A
B
B
C
C
C
C
D
D
D
D
D
D
E
E
E
E
E
E
E
E
Text
Red nodes are ignored because of infeasible solution through them
Weight = 8.98
Value = $235
Weight = 8.98
Value = $235
Black nodes are ignored because the best solution through them is worst than the current solution

A	2;	$40
B	3.14	$50
C	1.98	$100
D	5	$95
E	3	$30

For Other Example refer: https://www.javatpoint.com/branch-and-bound

13. Where is Dynamic programming used in 0/1 knapsack

➔ The basic idea of Knapsack dynamic programming is to use a table to store the solutions of solved subproblems. If you face a subproblem again, you just need to take the solution in the table without having to solve it again. Therefore, the algorithms designed by dynamic programming are very effective.

To solve a problem by dynamic programming, you need to do the following tasks:

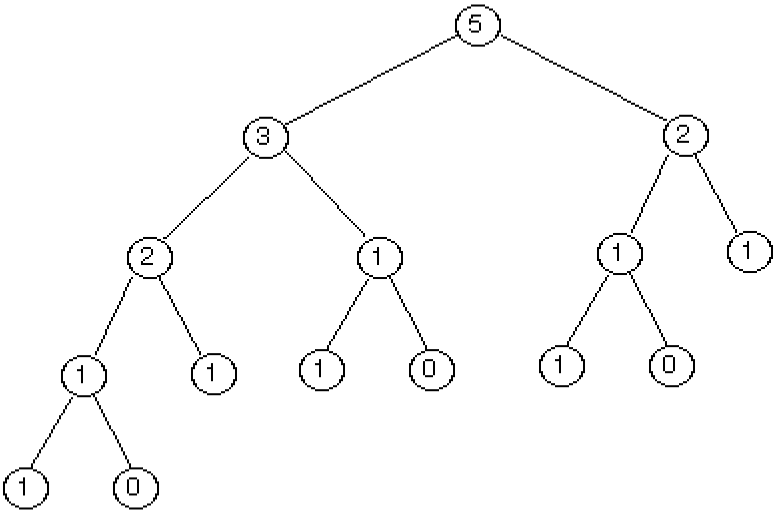
- Find solutions of the smallest subproblems.
- Find out the formula (or rule) to build a solution of subproblem through solutions of even smallest subproblems.
- Create a table that stores the solutions of subproblems. Then calculate the solution of subproblem according to the found formula and save to the table.
- From the solved subproblems, you find the solution of the original problem.

14. When Greedy is efficient and when Dynamic programming is efficient

➔ As a general rule, dynamic programming is better than greedy programming when you are solving problems that involve large numbers of variables or when you are solving problems that involve a lot of data. On the other hand, greedy programming is better than dynamic programming when you are solving problems that involve small numbers of variables or when you are solving problems that involve a lot of data. A dynamic programming algorithm is a powerful tool that can be used to solve problems more efficiently than with other algorithms.

15. Draw recursion tree for Fibonacci series

➔



16. What is backtracking, Explain in N-queens

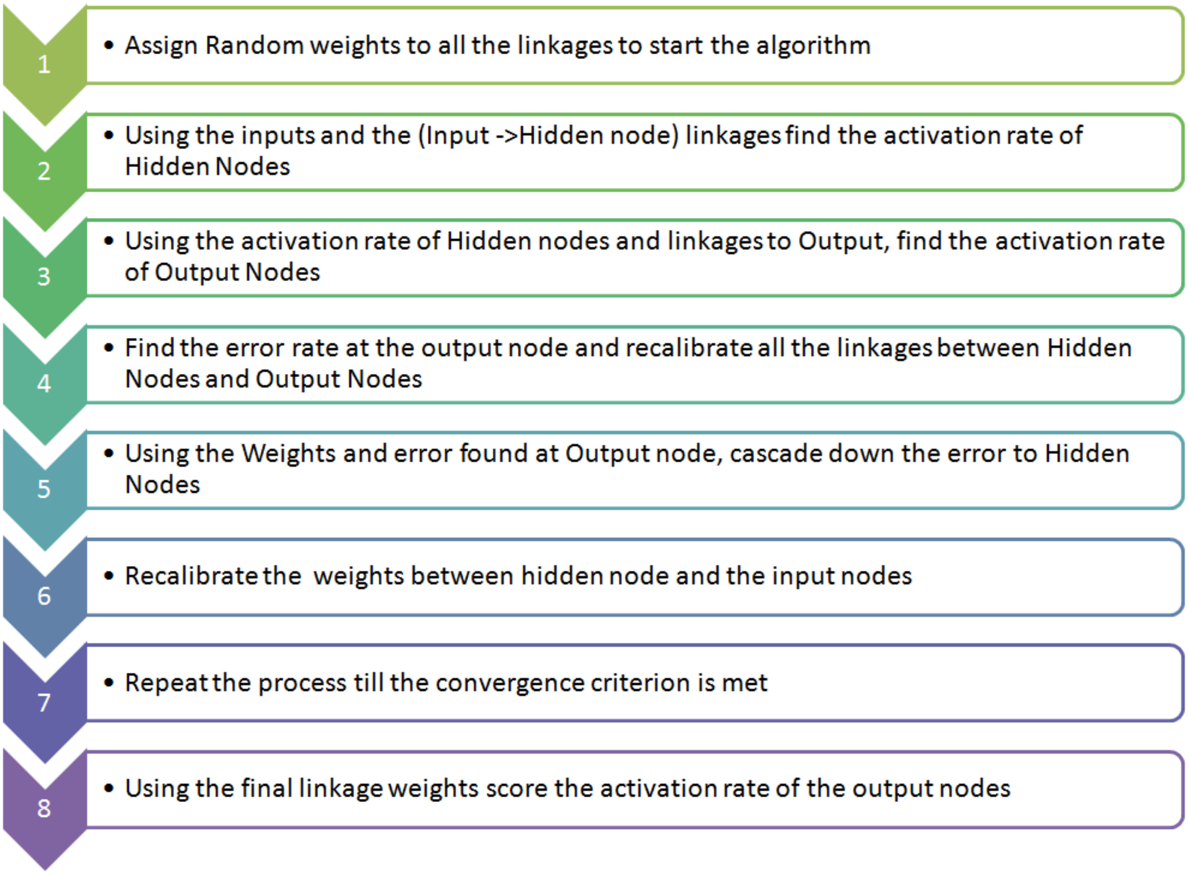
➔ In backtracking, we start with one possible move out of many available moves. We then try to solve the problem.

If we are able to solve the problem with the selected move then we will print the solution. Else we will backtrack and select some other move and try to solve it.

If none of the moves works out we claim that there is no solution for the problem.

The N Queen is the problem of placing N chess queens on an N×N chessboard so that no two queens attack each other. The idea is to place queens one by one in different columns, starting from the leftmost column. When we place a queen in a column, we check for clashes with already placed queens. In the current column, if we find a row for which there is no clash, we mark this row and column as part of the solution. If we do not find such a row due to clashes, then we backtrack and return false.

17. How ANN works



1 • Assign Random weights to all the linkages to start the algorithm

2 • Using the inputs and the (Input ->Hidden node) linkages find the activation rate of Hidden Nodes

3 • Using the activation rate of Hidden nodes and linkages to Output, find the activation rate of Output Nodes

4 • Find the error rate at the output node and recalibrate all the linkages between Hidden Nodes and Output Nodes

5 • Using the Weights and error found at Output node, cascade down the error to Hidden Nodes

6 • Recalibrate the weights between hidden node and the input nodes

7 • Repeat the process till the convergence criterion is met

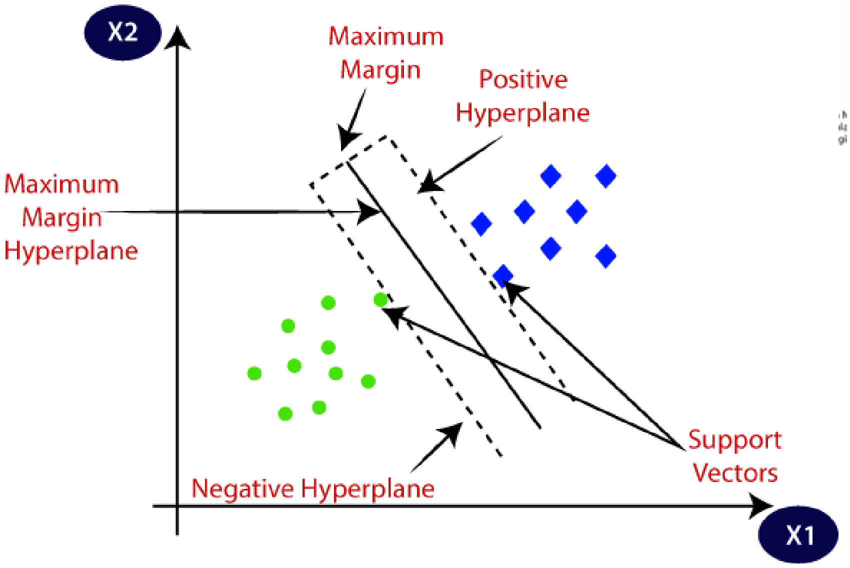
8 • Using the final linkage weights score the activation rate of the output nodes

➔

18. How SVM works

➔ A simple linear SVM classifier works by making a straight line between two classes. That means all of the data points on one side of the line will represent a category and the data points on the other side of the line will be put into a different category. This means there can be an infinite number of lines to choose from.

SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine. Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane:



19. What is kernel in SVM, what are the types, Explain each

➔

20. How KNN differs from K-means

➔ **Kernel Function** is a method used to take data as input and transform it into the required form of processing data. "Kernel" is used due to a set of mathematical functions used in Support Vector Machine providing the window to manipulate the data. So, Kernel Function generally transforms the training set of data so that a non-linear decision surface is able to transform to a linear equation in a higher number of dimension spaces.

- **Gaussian Kernel:** It is used to perform transformation when there is no prior knowledge about data.

  $$k(x, y) = \exp\left(-\frac{||x - y||^2}{2\sigma^2}\right)$$

- **Gaussian Kernel Radial Basis Function (RBF):** Same as above kernel function, adding radial basis method to improve the transformation.

  $$k(\mathbf{x_i}, \mathbf{x_j}) = \exp(-\gamma||\mathbf{x_i} - \mathbf{x_j}||^2)$$

- **Sigmoid Kernel:** this function is equivalent to a two-layer, perceptron model of the neural network, which is used as an activation function for artificial neurons.

  $$k(x, y) = \tanh(\alpha x^T y + c)$$

- **Polynomial Kernel:** It represents the similarity of vectors in the training set of data in a feature space over polynomials of the original variables used in the kernel.

  $$k(\mathbf{x_i}, \mathbf{x_j}) = (\mathbf{x_i} \cdot \mathbf{x_j} + 1)^d$$

21. What is PCA

➔ Principal Component Analysis is an unsupervised learning algorithm that is used for the dimensionality reduction in machine learning. It is a statistical process that converts the observations of correlated features into a set of linearly uncorrelated features with the help of orthogonal transformation. These new transformed features are called the **Principal Components**.

22. What is seaborn

➔ Seaborn is **a library for making statistical graphics in Python**. It builds on top of matplotlib and integrates closely with pandas data structures.

23. What is Sklearn

➔ Scikit-learn is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support-vector machines

24. What is train test split

➔ The train test validation split is a technique for partitioning data into training, validation, and test sets. It is a model validation procedure that reveals how your model performs on new data.

25. What is random state in train test split

➔ The random state hyperparameter in the train_test_split() function **controls the shuffling process**. With random_state=None , we get different train and test sets across different executions and the shuffling process is out of control. With random_state=0 , we get the same train and test sets across different executions.

26. What is cross K-fold validation

➔ Cross-validation is **a resampling procedure used to evaluate machine learning models on a limited data sample**. The procedure has a single parameter called k that refers to the number of groups that a given data sample is to be split into.

27. How random forest regression works

➔ Random forest is an ensemble of decision trees. This is to say that many trees, constructed in a certain "random" way form a Random Forest.

- Each tree is created from a different sample of rows and at each node, a different sample of features is selected for splitting.
- Each of the trees makes its own individual prediction.
- These predictions are then averaged to produce a single result.
- The averaging makes a Random Forest better than a single Decision Tree hence improves its accuracy and reduces overfitting.
- A prediction from the Random Forest Regressor is an average of the predictions produced by the trees in the forest.

28. What is regression

➔ Regression is a method to determine the statistical relationship between a dependent variable and one or more independent variables. The change independent variable is associated with the change in the independent variables. This can be broadly classified into two major types. Linear Regression. Logistic Regression.

29. What are classification algorithm

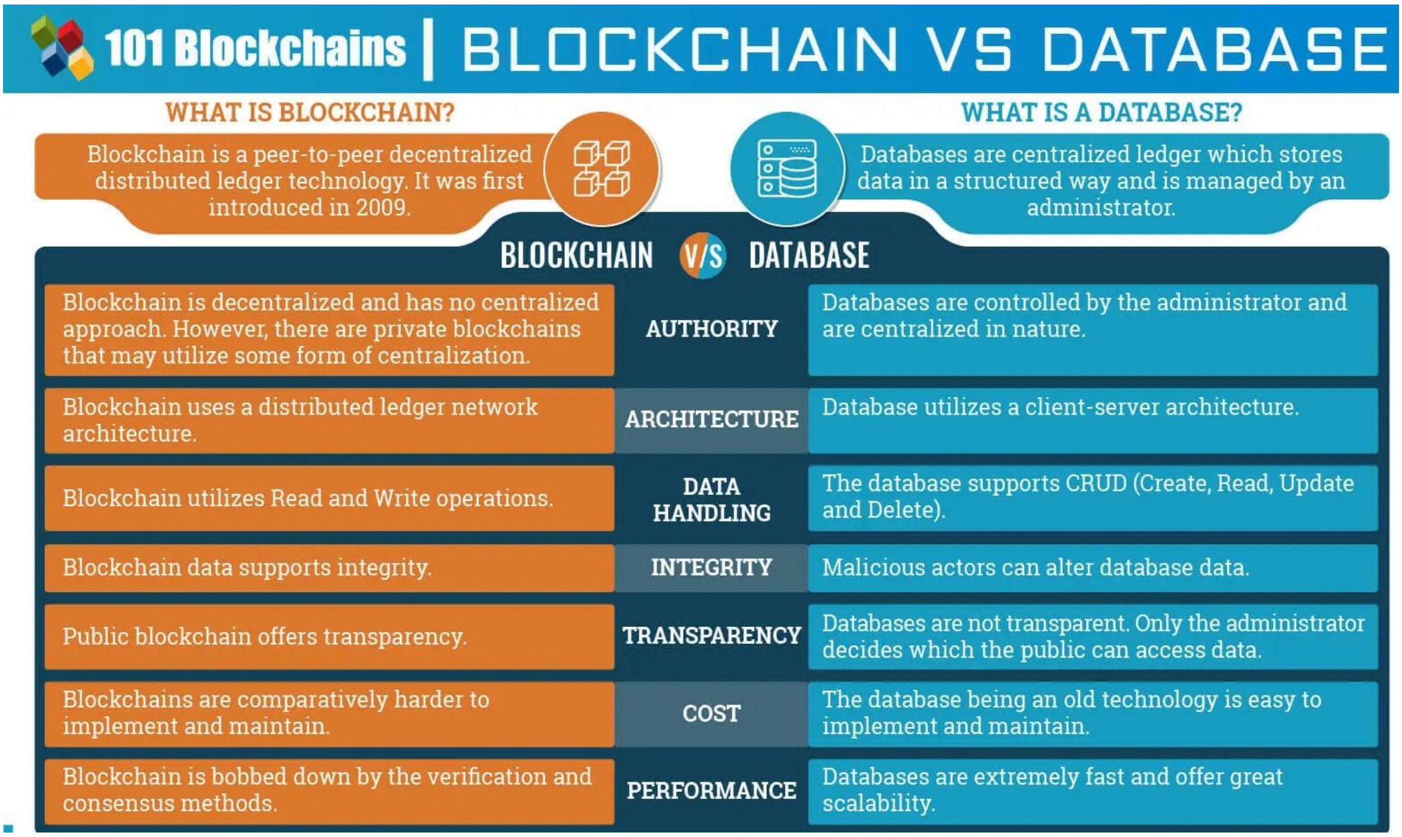
➔ The Classification algorithm is a Supervised Learning technique that is used to identify the category of new observations on the basis of training data.

30. What is blockchain

➔ A blockchain is a decentralized, distributed and public digital ledger that is used to record transactions across many computers so that the record cannot be altered retroactively without the alteration of all subsequent blocks and the consensus of the network.

31. How blockchain differs from Data base

➔



101 Blockchains | BLOCKCHAIN VS DATABASE

WHAT IS BLOCKCHAIN?

Blockchain is a peer-to-peer decentralized distributed ledger technology. It was first introduced in 2009.

WHAT IS A DATABASE?

Databases are centralized ledger which stores data in a structured way and is managed by an administrator.

BLOCKCHAIN	V/S	DATABASE
Blockchain is decentralized and has no centralized approach. However, there are private blockchains that may utilize some form of centralization.	AUTHORITY	Databases are controlled by the administrator and are centralized in nature.
Blockchain uses a distributed ledger network architecture.	ARCHITECTURE	Database utilizes a client-server architecture.
Blockchain utilizes Read and Write operations.	DATA HANDLING	The database supports CRUD (Create, Read, Update and Delete).
Blockchain data supports integrity.	INTEGRITY	Malicious actors can alter database data.
Public blockchain offers transparency.	TRANSPARENCY	Databases are not transparent. Only the administrator decides which the public can access data.
Blockchains are comparatively harder to implement and maintain.	COST	The database being an old technology is easy to implement and maintain.
Blockchain is bobbed down by the verification and consensus methods.	PERFORMANCE	Databases are extremely fast and offer great scalability.

32. What is mining

➔ Mining is the process that Bitcoin and several other cryptocurrencies use to generate new coins and verify new transactions. It involves vast, decentralized networks of computers around the world that verify and secure blockchains – the virtual ledgers that document cryptocurrency transactions. In return for contributing their processing power, computers on the network are rewarded with new coins.

33. What is solidity

➔ Solidity is an object-oriented programming language for implementing smart contracts on various blockchain platforms, most notably, Ethereum. It was developed by Christian Reitwiessner, Alex Beregszaszi, and several former Ethereum core contributors. Programs in Solidity run on Ethereum Virtual Machine.

34. how solidity is different from java, c++ etc

➔ Solidity is an object-oriented and statically-typed programming language that was designed to allow developers to create smart contracts.

While Java has many complex and simple data structures built-in, **in Solidity there are only two of them: mapping and array**. The first one is a simplified version of Java map, meaning it can store key-value pairs and retrieve the value stored under a given key, but not much else.

Writing in Solidity requires paying attention to many more details than when writing in Java.

Array access syntax looks like in Java but declaration syntax is reversed — int8[][5] creates 5 dynamic arrays of bytes. Unfortunately only statically sized arrays can be returned by functions — there is no way to return a dynamically sized one. Also, multi-dimensional dynamic arrays cannot be created which means that declaring an array of strings (which are dynamically sized arrays of chars) is not possible.

35. What is memory in solidity

➔ Much like RAM, Memory in Solidity is **a temporary place to store data** whereas Storage holds data between function calls. The Solidity Smart Contract can use any amount of memory during the execution but once the execution stops, the Memory is completely wiped off for the next execution.

36. What is virtual machine

➔ A VM is a virtualized instance of a computer that can perform almost all of the same functions as a computer, including running applications and operating systems. Virtual machines run on a physical machine and access computing resources from software called a hypervisor.

37. What is Meta mask and why it is used

➔ MetaMask is a **software cryptocurrency wallet used to interact with the Ethereum blockchain**. It allows users to access their Ethereum wallet through a browser extension or mobile app, which can then be used to interact with decentralized applications.